Proceedings of the 2008 IEEE/ASME
International Conference on Advanced Intelligent Mechatronics
July 2 - 5, 2008, Xi'an, China

Intelligence Comparison between Fish and Robot using Chaos and Random

Jun Hirao and Mamoru Minami

***Abstract*— In this paper we tackle a Fish-Catching task under a visual feedback hand-eye robotic system with a catching net. As the time of tracking and catching process flows, the fish can somewhat get accustomed to the net motion pattern and gradually find out new strategies on how to escape from the bothering net. For the sake of such innate ability being widely existed in animal behavior, the catching operation becomes tough and some effective intelligent method needs to be conceived to go beyond the fish intelligence. The purpose of this paper is to construct intelligent system to be able to exceed the fish intelligence in order to track and catch the fish successfully. Then we embed chaotic and random motion into the net motion to realize a kind of robotic intelligence, and we shown the chaotic and random net motion is effective to overcome the fish escaping strategies. The effectiveness of the chaotic and random motion is confirmed through successive fish catching experiment.**

I. INTRODUCTION

In recent years, visual tracking and servoing in which visual information is used to direct the end-effector of a manipulator toward a target object has been studied in some researches [1],[2]. A new trend of machine intelligence [3] that differs from the classical AI has been applied intensively to the field of robotics and other research areas like intelligent control system. Typically, the animal world has been used conceptually by roboticists as a source of inspiration for machine intelligence. For the purpose of studying animal behavior and intelligence, the model of interaction between animals and machines is proposed in researches like [4]. In our previous research, the fish emotional behavior has also been examined and the robot with adaptive ability to react to the fish status is conceived. Another crucial characteristic of machine intelligence is that the robot should be able to use input information from sensor to know how to behave in a changing environment and furthermore can learn from the environment for safety like avoiding obstacle. As known universally that the robot intelligence has reached a relatively high level, still the word intelligence is an abstract term, so the measurement of the intelligence level of a robot has become necessary. A practical and systematic strategy for measuring machine intelligence quotient (MIQ) of human-machine cooperative systems is proposed in [5].

In our system, we will evaluate the intelligence degree between fishes and the robot by Fish-Catching operation. We can declare that the system combined with chaos be smarter than the fish when the robot can beat the fish by catching it successfully even after the fish finds out some escaping strategy. As we did not find the research about the intelligence comparison between animal and robot, we mainly dedicate ourselves to constructing a smart system that is more intelligent than the fish. We consider that the competitive relation can be very meaningful as one way to discuss robotic intelligence. So we not only employ the inspiration of animal behavior for robot intellectualization, we can also conceive a robot that can exceed the animal intelligence. By evolutionary algorithms [6], Visual Servoing and Object Recognizing based on the input image from a CCD camera mounted on the manipulator has been studied in our laboratory(Fig.1) [7], and we succeeded in catching a fish by a net attached at the hand of the manipulator based on the real-time visual recognition under the method of Gazing GA [8] to enhance the real-time searching ability.

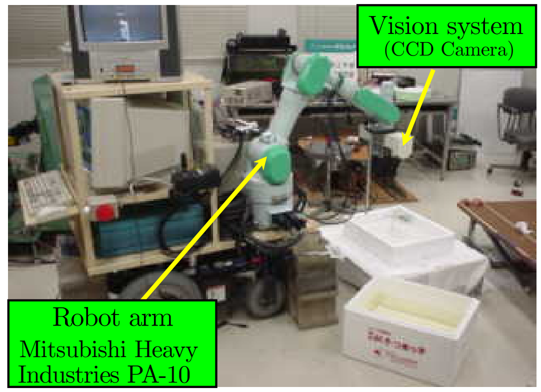


Fig. 1. Fish Catching system PA10

We have learned that it is not effective for fish catching to simply pursue the current fish position by visual servoing with velocity feedback control. Actually, the effective tracking can be impossible because the fish can sometimes alter motion pattern suddenly under some emotional reasons of fear or the fish can take some strategy to try to get rid of the bothering net that keeps chasing it. Those behaviors are thought to be caused by emotional factors and they can also be treated as a kind of innate fish intelligence, even though not in a high level. Based on the fish behavior observation in the real Fish-Catching experiment, the fish mostly swims stick to the pool edge for avoiding the net after being caught several times. That is a serious problem for the Fish-Catching task because when the fish only swims in the corner it is prohibited for the net to enter into the corner in Fish-Catching operation. That shows the system is not intelligent enough, so effective method is expected to be conceived in

Jun Hirao is with Graduate school of Engineering, University of Fukui, Fukui, 910–8507, Japan hirao@rc.his.fukui-u.ac.jp
Mamoru Minami is with Faculty of Engineering, University of Fukui, Fukui, 910–8507, Japan minami@rc.his.fukui-u.ac.jp

order to cope with the fish escaping strategy. While observing the fishes' adapting behavior to escape in the competitive relations with the robot, we found that we can define a "Fish's Intelligent Quotient"(FIQ) representing decreasing velocity of fish number caught by the net through continuous catching/releasing operation. Through this measure we will compare the innate intelligence of the fish and the artificial intelligence of the robot.

In this paper we adopt the chaos model obtained from signal transfer in cell structure [9],[10]. We embed chaos into the Robot Dynamics in order to supplement the deficiency of our Fish-Catching system, because intelligent composite motion control [11] becomes crucial in the catching fish process. The Chaotic motion will be added to increase the possibility of catching fish according to the fish motion state and we can call that motion adaptive ability that is being researched in our lab [12]. We have tried a new strategy to make the system smart enough to exceed the fish intelligence. Moreover, we embed random motion into the net motion in order to compare with chaotic motion, because random and chaos have relations of intersection, that means it'll be meaningful to compare random and chaos in Catching-Fish experiments.

II. RANDOM AND CHAOS

A random number is unpredictable. It seems that it is impossible to express the random number by computer program, because computer program is just able to output a sequence of numbers by prescribed programs, resulting in the output number be essentially predictable. Actually, the random number generation routine in computer is called pseudorandom number, it is not real random number with genuine unpredictability. That means the pseudorandom number is predictable. A function to generate a "random number" prepared in a standard language like "C++" is based on the linear congruential method almost without exception, producing pseudorandom number of cyclic oscillation with huge cycle. This method is proposed by Lehmer, D.H. around 1948, and it's so easy and efficient for generating pseudorandom-numbers, named "linear congruential method" with the following recurrence formula [13].

$$X_n = aX_{n-1} + c \pmod{M}, n \geq 1 \tag{1}$$

This equation can output integer pseudorandom-numbers sequence $X_0, X_1, X_2, \cdots$. The M is called modulus of congruence expression. a and c are positive integers. a is called multiplier, c is called increment. So, the remainder value, coming from $aX_{n-1} + c$ divide by M, is set to X_n. In eq.(1) there is a period, this period is no larger than M. If M, a, and c are chosen well combination, the maximum cycle M can be obtained. In the case of the maximum period, all the integer numbers not smaller than 0 and not larger than $M - 1$ appear in somewhere. No mater X_0, it becomes the same sequence of numbers after all only by a sequence of numbers beginning from there, and this is a periodic function.

Chaos has the character of unpredictability. This happens by negligible differences of initial positions bear unpredictable huge difference between the solved trajectories. This means that deterministic equation of chaos can generate unpredictability. The Bernoulli shift is mentioned as a typical example that realizes the character of the chaos. The Bernoulli shift is expressed by considering the variable X_i is a real number and by substituting $a = 2, c = 0, M = 1$ into (1).

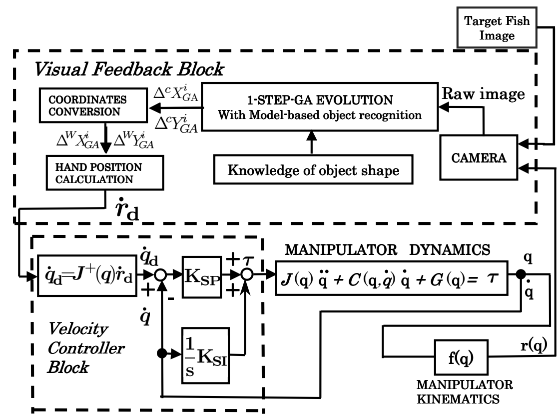


Fig. 2. Block diagram of the controller

$$X_n = 2X_{n-1} \pmod{1}, n \geq 1 \tag{2}$$

That is, chaos and pseudorandom numbers can be generated by the same equation. As mentioned above, we consider that chaos and random numbers have relations of intersection.

III. FISH TRACKING AND CATCHING

The problem of recognition of a fish and detection of its position/orientation is converted to a searching problem of $\boldsymbol{r}(t) = [x(t), y(t)]^\top$ in order to maximize $F(\boldsymbol{r}(t))$, where $F(\boldsymbol{r}(t))$ represents correlation function of a new image and matching model to a fish at time t. $F(\boldsymbol{r}(t))$ is used as a fitness function of GA [8]. To recognize a target in a dynamic image input by video rate, 33 [fps], the recognition system must have real-time nature, that is, the searching model must converge to the fish in the successively input raw images. An evolutionary recognition process for dynamic images is realized by such method whose model-based matching by evolving process in GA is applied at least only one time to one raw image input successively by video rate. We named it as "1-Step GA" [7]. When the converging speed of the model to the target in the dynamic images should be faster than the swimming speed of the fish in the dynamic images, then the position indicated by the highest genes represent the fish's position in the successively input images in real-time. We have confirmed that the above time-variant optimization problem to solve $\boldsymbol{r}(t)$ maximizing $F(\boldsymbol{r}(t))$ could be solved by "1-Step GA". $\boldsymbol{r}(t) = [x(t), y(t)]^\top$ represents the fish's position in Camera Frame whose center is set at the center of catching net, then $\boldsymbol{r}(t)$ means position deviation from net to Fish, means $\boldsymbol{r}(t) = \Delta\boldsymbol{r}(t)$

The desired hand velocity at the i-th control period $\dot{\boldsymbol{r}}_d^i$ is calculated as

$$\dot{\boldsymbol{r}}_d^i = \boldsymbol{K}_P \Delta\boldsymbol{r}^i + \boldsymbol{K}_V(\Delta\boldsymbol{r}^i - \Delta\boldsymbol{r}^{i-1}) \tag{3}$$

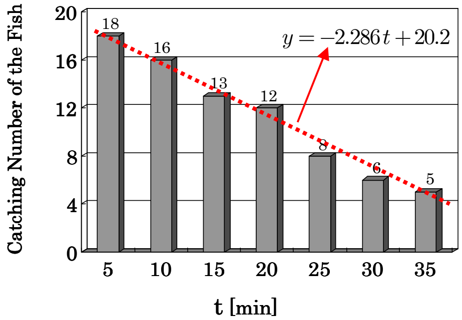
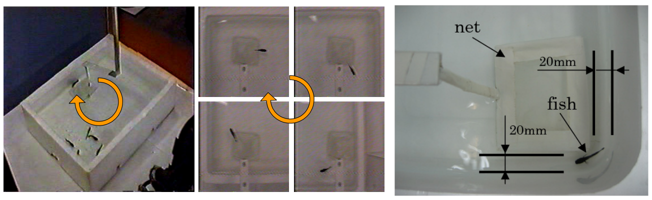


Fig. 3. Result of catching number only sovoing



(a) Motion (1) of a fish (b) Motion (2) of a fish

Fig. 4. Fish motion

where $\Delta \boldsymbol{r}^i$ denotes the servoing position error detected by 1-Step GA [7]. $\boldsymbol{K}_P$ and $\boldsymbol{K}_V$ given are positive definite matrix to determine PD gain. Now we add chaos items to (3) above, and we also need to redefine the meaning of $\dot{\boldsymbol{r}}_d^i$.

The simple PD servo control method given by (3) is modulated to combine a visual servoing and chaos net motion into the controller as follows,

$$\Delta \boldsymbol{r}^i = k_1 \cdot \Delta \boldsymbol{r}^i_{fish} + k_2 \cdot \Delta \boldsymbol{r}^i_{chaos} \tag{4}$$

Here $\Delta \boldsymbol{r}^i_{fish} = \begin{bmatrix} \Delta x^i_{fish} & \Delta y^i_{fish} \end{bmatrix}$, and $\Delta \boldsymbol{r}^i_{chaos} = \begin{bmatrix} \Delta x^i_{chaos} & \Delta y^i_{chaos} \end{bmatrix}$., where $\Delta \boldsymbol{r}^i_{fish} = \Delta \boldsymbol{r}^i_{chaos}$ denotes a chaotic oscillation in $x - y$ plane. Therefore the hand motion pattern can be determined by the switch value k_1 and k_2. $k_1 = 1$ and $k_2 = 0$ indicate visual servoing, and $k_1 = 0$ and $k_2 = 1$ indicate the net will do chaotic motion under certain condition satisfied either to lure the fish to come out of the corner or threaten the fish. The desired joint variable $\dot{\boldsymbol{q}}_d$ is determined by inverse kinematics from $\dot{\boldsymbol{r}}_d$ by using the Jacobian matrix $\boldsymbol{J}(\boldsymbol{q})$, and is expressed by

$$\dot{\boldsymbol{q}}_d = \boldsymbol{J}^+(\boldsymbol{q})\dot{\boldsymbol{r}}_d \tag{5}$$

where $\boldsymbol{J}^+(\boldsymbol{q})$ is the pseudoinverse matrix of $\boldsymbol{J}(\boldsymbol{q})$. The robot used in this experimental system is a 7-Link manipulator, Mitsubishi Heavy Industries PA-10 robot.

IV. PROBLEM OF FISH-CATCHING

In order to check the system reliability in tracking and catching process, we kept catching several fishes continuously for 35 minutes with condition of $k_1 = 1$ and $k_2 = 0$ through out this experiment. We released 8 fishes (length is about 40[mm]) in the pool in advance, and once the fish got caught it would be released to the same pool at once. The result of this experiment is shown in Fig.3, in which vertical axis represents the number of fishes caught in successive 5 minutes and horizontal axis represents the catching time. We had expected that the capturing operation would become smoother as time flew on consideration that the fish may get tired. But to our astonishment, the number of fishes caught decreased gradually.

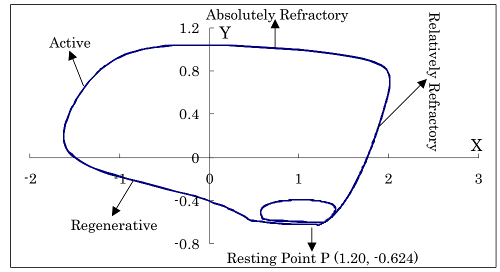


Fig. 5. Chaos trajectory

The reason of decreased catching number lies in the fish learning ability or emotional factor stated before. For example, the fish could learn how to run away around the net shown in Fig.4(a) by circular motion with keeping constant distance from the net. Also, the fish can keep staying within the clearance between the edge of the pool and the net shown in Fig.4(b) where it is prohibited for the net to enter. To evaluate numerically how fast the fish can learn to escape the net, we adapted Linear Least-Square approximation to the fish-catching decreasing tendency, resulting in $y = -2.286t + 20.2$ as shown in Fig.3. The decreasing coefficient -2.286 represents adapting or learning velocity of the fishes as a group when the fishes' intelligence is evaluated based on robotic performance given as a standard. We named the coefficient as "*Fish's Intelligence Quotient*"(FIQ). The larger minus value means high intelligence quotient, zero does equal, and plus does less intelligence than robot's.

To overcome the fishes' intelligence, more intelligent robotic system needs to track and catch the fish effectively, in other words it comes to the problem on how to use the item $\Delta \boldsymbol{r}^i_{chaos}$ in (4) effectively to exceed the fish intelligence.

V. CHAOS BVP MODEL

The phenomenon of chaotic motion can be frequently seen in living body. For example when you stare at one point still without blinking, it appears that your eyeballs have stopped motion somewhat, but in fact the eyeballs are still making an infinitesimal movement that is called chaotic vibration. FitsHugh proposed BVP model in order to imitate the chaotic motion of nerves excitement in the potential plane [9]. The BVP equation can be deduced from the following differential equation.

$$\ddot{x} + c(x^2 - 1)\dot{x} + x = 0 (c > 0) \tag{6}$$

Here $\dot{x}$ signifies the differential of x with respect to time, and we take a linear transformation as follow:

$$F(x) = \int_0^x f(u)\, du \tag{7}$$

Given the supposed (7) and the following transformed equation of (6):

$$\ddot{x} + f(x)\dot{x} + g(x) = 0 (c > 0) \tag{8}$$

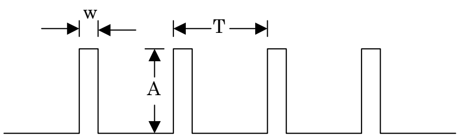


Fig. 6. The stimulus signal

TABLE 1

	A	T(ms)	W(ms)	type of response
(a)	-0.8	100	3	2 stimulus : 1 response
(b)	-0.91	100	3	5 stimulus : 4 responses
(c)	-0.95	100	3	1 stimulus : 1 response
(d)	-0.87	50	3	chaotic response

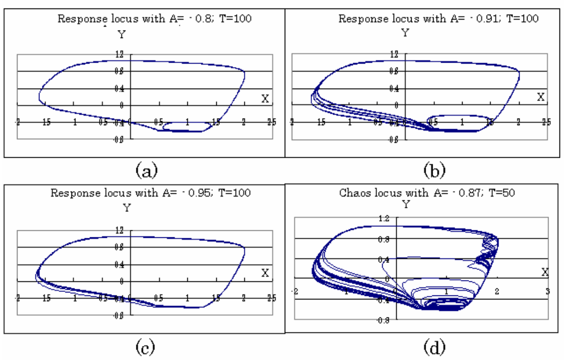


Fig. 7. The BVP response to stimulus

We can obtain the following two differential equations related with two variants x and y.

$$\dot{x} = y - F(x) = y - \int_0^x c(u^2 - 1)\, du \tag{9}$$

$$\dot{y} = -g(x) = -x \tag{10}$$

The full BVP model form can be finally acquired by adding a stimulus item z and the BVP equation is acquired as follow:

$$\begin{aligned} \dot{x} &= c(x - \frac{x^3}{3} + y + z) \\ \dot{y} &= -\frac{x + by - a}{c} \end{aligned} \tag{11}$$

Here we will give biological definition about x and y arisen from BVP differential equation. The item x denotes the value of reversal sign of membrane voltage in cell and y signifies the refractory nature. The item z denotes the outer stimulus signal. Parameters a, b and c are confined as follow based on [10]:

$$1 - \frac{2b}{3} < a < 1, \quad 0 < b < 1, \quad b < c^2 \tag{12}$$

Fig.5 shows one example about the solution trajectory of BVP differential equation in x-y potential plane, and the nerve exciting process can also be obviously observed from this figure. In this chaotic locus, we adopt pulse signal as the stimulus. When proper stimulus signal shown in Fig.6 comes, nerve excitement can happen. The respond process will start from Resting Point P, then pass through Regenerative part, Active part, Absolutely Refractory part, then Relatively Refractory part, and finally return back to Resting Point P

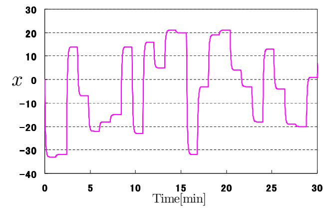


Fig. 8. Random x

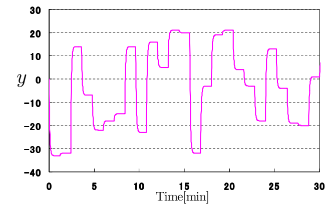


Fig. 9. Random y

again. In other words, the responsive trajectory has such characteristic that although the nerve cell is in a state of stillness originally, it will get excited once accepting a proper instant pulse signal and return to the stationary state in the end. The coordinate of resting point P is (1.20,-0.624) as one characteristic of BVP solution.

In order to apply chaos to the current intelligent system appropriately, the relationship between chaotic response and stimulus need to be examined. One proposed method called Rungekutta that has an relatively optimal performance in solving differential equation has been adopted to the BVP differential equation solution procedure. Now we merely let parameters A and T in BVP equation change with item W fixed to analyze the respond pattern. In the actual simulation by C program, if the amplitude A is strong enough and the T is long enough there will be a response once stimulus signal comes. On the contrary, if amplitude A is set weak enough or T short enough, there will not be any respond to stimulus no matter how frequent the stimulus comes. The respond pattern with different A and T is shown in Fig.7. The parameters of stimulus signal and various responsive types are shown in Table 1.

VI. RANDOM MODEL

Random numbers are generated by using the random number generation routine of Borland C++ Builder5. Here, we define random referring (1) as follow:

$$X_n = Random_x(X_{n-1}) \tag{13}$$

$$Y_n = Random_y(Y_{n-1}) \tag{14}$$

Where n means the time of $\Delta t \cdot n$ and n is sequence number of random generated by above function. Then we have new random sequence of $X_0, X_1, X_2, \cdots$ and $Y_0, Y_1, Y_2, \cdots$. What is the problem here to give this random sequence to the net motion, is the robot motion controller may be oscillative because the random sequence motion demands mean a kind of successive step responses. To avoid this problem derived from robot‘s dynamics, we thought the random step motion should be deformed by filtering by first order differential equations, for example the hand x, y positions during the time period $n \cdot T \le t \le (n+1) \cdot T$ are given as,

$$\dot{x} + x = X_n \tag{15}$$

$$\dot{y} + y = Y_n \tag{16}$$

with initial value of x at time $t = n \cdot \Delta t$ is given by the previous filtering result of $x(\Delta t \cdot n)((n-1) \cdot T \le t \le n \cdot T)$, and y is also. Here, the item T denotes the time cycle that updates random numbers and here is set to

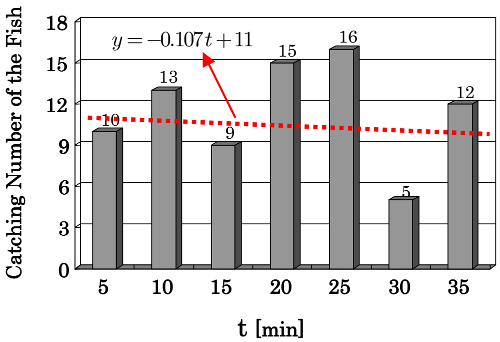


Fig. 10. Result of catching number using Chaos

be $T = 1.2[s]$. We use one method called Rungekutta in solving differential equation, which has been adopted for solving these differential equation. Fig.8 and Fig.9 shows the example about the solution of (15) and (16). Vertical axis represents x and y, horizontal axis represents the time.

The deformed random trajectories $x(t), y(t)$ are used to define $\Delta \boldsymbol{r}^i_{random}$ as $\Delta \boldsymbol{r}^i_{random} = \left[\ x(t), y(t)\ \right]$ and by replacing a eq.(4) as follows, random is used on the same conditions as the time of chaos.

$$\Delta \boldsymbol{r}^i = k_1 \cdot \Delta \boldsymbol{r}^i_{fish} + k_2 \cdot \Delta \boldsymbol{r}^i_{random} \tag{17}$$

VII. CATCHING-FISH EXPERIMENT

A. Chaos Motion used

In order to examine whether the new system inserted with intelligent emotion imitation derived from chaos can be more effective, we have tried the catching-fish experiment. The idea is to let the time when the fish threatened in pool corner wants to swim out coincide with the time when the net begins to perform chaotic motion in a probabilistic way. So outer situations to the animal to determine its next action can be thought to correspond with the stimulus signal to the BVP chaos model. In other words, we try to imitate the animal judging pattern (mind) in the chaotic way. By this way, it is considered that the time when the fish feels like getting out of the corner can match the time when chaotic motion is to be taken. Then the fish tends to be deceived that the scaring net is going further away and it can be caught by the net immediately once it swims out of the corner.

In order to check whether the new proposed catching fish system is more effective than the original one, we also kept catching 8 fishes in pool continuously under the same condition as the catching-fish experiment in Fig.3. We recorded the catching number of fishes every 5 minutes. The catching fish number kept decreasing in the former experiment shown in Fig.3. But after we embedded the chaotic motion to net movement this time and the catching number of fish does not go down as shown in Fig.10. Although the fish-catching number is somewhat uncertain with time, the average number is about 11 and the decreasing tendency has stopped. It had been thought the number would apparently decrease due to fish innate intelligence, but this experiment result in Fig.10 is satisfying because of the chaotic motion the FIQ has varied from -2.286 in Fig.3 to -0.107 in Fig.10. In other words, the chaotic motion have compensated the problem of fish escaping ability to escape from the catching-net.

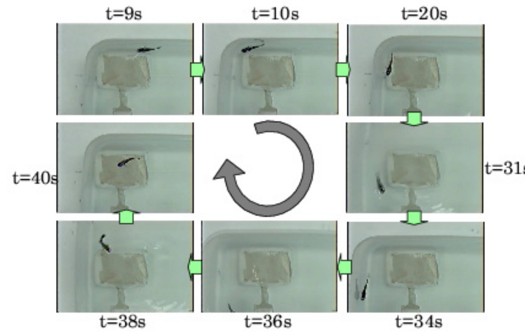


Fig. 11. The catching fish process by use of chaos

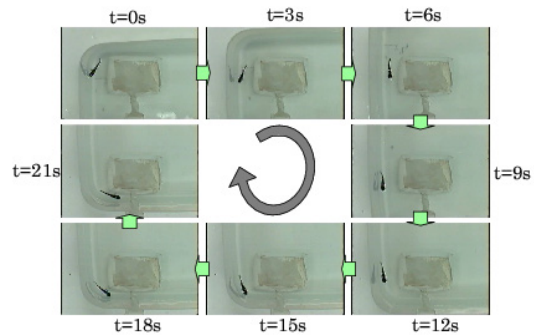


Fig. 12. The catching fish process without chaos

We took a close observation into the fish tracking and catching experiment using chaos. Here, we look at lasted for nearly $40s$ till the fish got caught successfully. During the first $9s$, the net mounted at hand sometimes moved round the pool regularly to find out the swimming fish and chased the fish once it appears in the camera vision view. After $9s$ passed, the fish began to swim slowly along the edge of the pool where it is forbidden for the net to enter. In order to observe clearly how the chaotic motion would act effectively towards that kind of fish escape strategy, we had taken a series of pictures during the remaining 31s shown in Fig.11. The picture with $t = 9s$ shows the fish began to swim slowly along the pool edge. After judging the current situation that the fish swam stick to the pool edge, the net took chaotic motion during time interval from $t = 10s$ to $t = 31s$. The pictures from $t = 31s$ to $t = 36s$ show the process of the fish swimming out of corner and the visual servoing system lost the fish who exceeded the hand-eye view area when $t = 36s$. When the condition that the fish becomes out of the current camera vision area is satisfied, the net is designated to move towards the position that possibly be the next fish place after the fish got lost on the basis of N.N. prediction result. So the fish fell into the vision span again and the net finally arrived at the place in front of the fish shown in picture with t=38s. The fish was finally caught at $t = 40s$ once it fell into the net range. The net is preset in the origin of this camera frame and it will be pulled up rapidly when the fish swims into the rectangular area 86×66[mm] round the net center.

In order to show the optimal performance with chaos adoption to the experimental system, we also take a series of

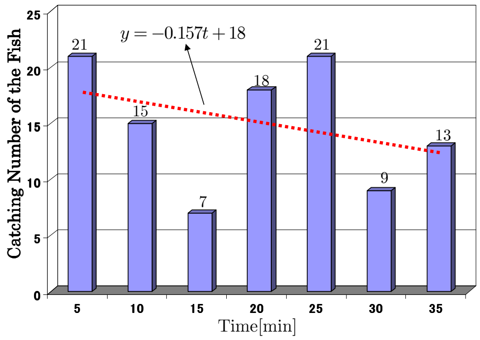


Fig. 13. Result of catching number using Random

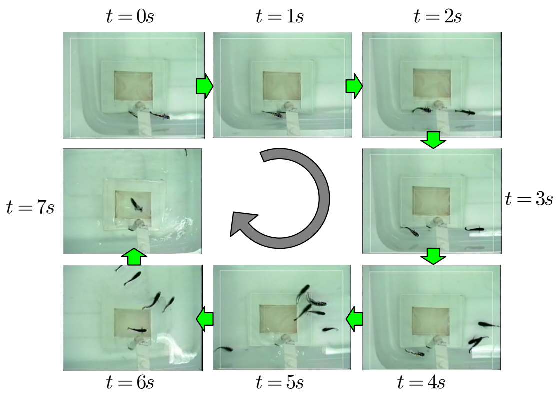


Fig. 14. The catching fish process by use of random

pictures every 3s when we do not use chaotic motion and the fish swims stick to the pool edge slowly. From the Fig.12 we can see the net only followed the fish outside the prohibited area. No matter how much time flows, the fish still stayed within the thin strip area along the pool edge without daring to swim out. So the result is quite different from Fig.11 with chaos use and the catching operation ended with failure this time.

As stated before, when the fish has finally gotten used to the net motion, it will be able to find out some escaping strategy to avoid the threatening net such as swimming within the pool corner where it is forbidden for the net to enter. So the catching number can become decreased as time flows.

B. Rondom Motion used

We showed that chaos and random has relations of intersection in Chapter 2. Then, random is used instead of chaos and we tried the catching-fish experiment. In this experiment, we also released 8 fishes into the pool and keep tracking, catching and releasing operation continuously for 35 minutes. We used a different 8-fish group compared with the using chaos experiment, with whose result shown in Fig.10. The result is shown in Fig.13, in which the horizontal axis represents time and vertical axis represents the Fish-Catching number in each 5 minute time span. After we embedded the random motion to net motion this time and the catching number of fish does not go down as shown in Fig.13 like the time of using chaos. When it compared using FIQ, the result with the more sufficient chaos was obtained because of the chaos motion FIQ is -0.107 and the random motion FIQ is -0.157. But the difference is not necessarily great at all. When catching fishes using chaos motion, the chaos motion is caught by luring out the fish from the corner of a pool. However, in the random motion, the way of capture of luring out and catching a fish was not seen. Fig.14 shows one example about the fish got caught successfully using random. The picture with $t = 0s$ shows the fish began to swim slowly along the pool edge. After judging the current situation that the fish swam stick to the pool edge, the net took random motion during time interval from $t = 0s$ to $t = 4s$. Here, other fish approach the net at t=4s, and the fish was recognized, then the fish was caught at t=7s once it fell into the net range. At the time of a random motion, decrease of the catching number of the fish was prevented by approaching the way of the fish by random motion in another fish to which it came by near the fish which has stayed in a corner, and catching other fish which are recognizing there and are not fish which have stayed in a corner.

VIII. CONCLUSION

We propose a new method so-called chaos embedded into the catching-net motion to cope with the fish learning ability to escape from net. We suggest intelligent system than the traditional one in order to exceed the intelligence of the fish and the effectively of the system is testified in real experiments. We embedded random motion into the net motion for compare with the chaotic motion. The experimental result has shown that the way to capture the fish is different by using chaos motion and random motion. But we cannot say the effectiveness of chaotic motion compare with random motion.

REFERENCES

[1] R. Kelly: "Robust Asymptotically Stable Visual Servoing of Planar Robots", IEEE Trans. Robot. Automat., vol.12, no.5, 1996, pp.759-766.
[2] P.Y. Oh, and P.K. Allen: "Visual servoing by partitioning degrees of freedom", IEEE Trans. Robot. Automat., vol.17, pp.1-17, Feb.2001
[3] T. Fukuda, and K. Shimojima: "Intelligent Control for Robotics", Computational Intelligence, 1995, pp.202-215.
[4] M. Bohlen: "A robot in a cage-exploring interactions between animals and robots", CIRA. 1999, pp.214-219.
[5] Hee-Jun Park, Byung Kook Kim, Kye Young Lim: "measuring the machine intelligence quotient (MIQ) of human-machine cooperative systems", IEEE Trans. vol.31, 2001, pp.89-96.
[6] M. Minami, H. Suzuki, J. Agbanhan, T. Asakura: "Visual Servoing to Fish and Catching Using Global/Local GA Search", Int. Conf. on Advanced Intelligent Mechatronics, Proc., 2001, pp.183-188.
[7] M. Minami, J. Agubanhan, and T. Asakura: "Manipulator Visual Servoing and Tracking of Fish using Genetic Algorithm", Int. J. of Industrial Robot, Vol.29, No.4, 1999, pp.278-289.
[8] Visual Servoing to catch fish Using Global/local GA Search Hidekazu Suzuki, Mamoru Minami IEEE/ASME Transactions on Mechatronics, Vol.10, Issue 3, 352-357 (2005.6)
[9] K. Aihara: "Chaos in Neural System", pp.126-151, 1993 (in Japanese).
[10] R. FitzHugh: "Impulses and physiological states in theoretical models of nerve membrane", Biophy.J.1, pp.445-466 (1961).
[11] M. Suzuki.: "Evolutionary acquisition of complex behaviors through intelligent composite motion control", Computational Intelligence in Robot. Automat. CIRA, vol.27-30, 2005, pp.231-236.
[12] M. Minami, T. Asakura and A. Tamamura: "Machine intelligence of a mobile manipulator to utilize dynamically interfered motion and nonlinear friction", WCCI. Congress on Evolutionary Computation, 2002, pp.1232-1237.
[13] William H. Press, Saul A. Tenkolsky, William T. Vetterling, Brian P. Flannery: "Numerical Recipes in C -The Art of Scientific Computing Second Edition-" PP.274-328, 2002, U.S.A.